

MATH 545: INTRODUCTION TO TIME SERIES ANALYSIS

WINTER 2025

DECOMPOSITION

We seek to decompose an observed series $\{Y_t\}$ as

$$Y_t = m_t + s_t + X_t$$

where

- m_t is a deterministic trend
- s_t is a seasonal component with period d
- X_t is a zero mean stationary process.

Smoothing: For a non-seasonal data series $\{x_t\}$, we may proceed by smoothing to remove non-linear trends:

- **Smoothing by averaging:** Suppose

$$\hat{m}_t = \frac{1}{2q+1} \sum_{i=-q}^q x_{t+i}$$

with $x_s = 1$ for $s < 1$ and $s > T$. More generally,

$$\hat{m}_t = \sum_{i=-q}^q a_i x_{t+i}$$

for constants $a_{-q}, \dots, a_q$.

- **Exponential Smoothing:** Suppose $\hat{m}_1 = x_1$, and for $t \geq 2$

$$\hat{m}_t = \alpha x_t + (1 - \alpha) \hat{m}_{t-1}$$

for $0 < \alpha < 1$. It follows that

$$\hat{m}_t = (1 - \alpha)^{t-1} x_1 + \sum_{j=1}^{t-2} \alpha (1 - \alpha)^j x_{t-j} + \alpha x_t$$

If α is near one, there is strong dependence on recent x values.

EXAMPLE: Suppose we have the process

$$X_t = m_t + E_t$$

for

$$m_t = 10 + 0.01t + 5 \exp\{-0.001(t - 100)^2\}$$

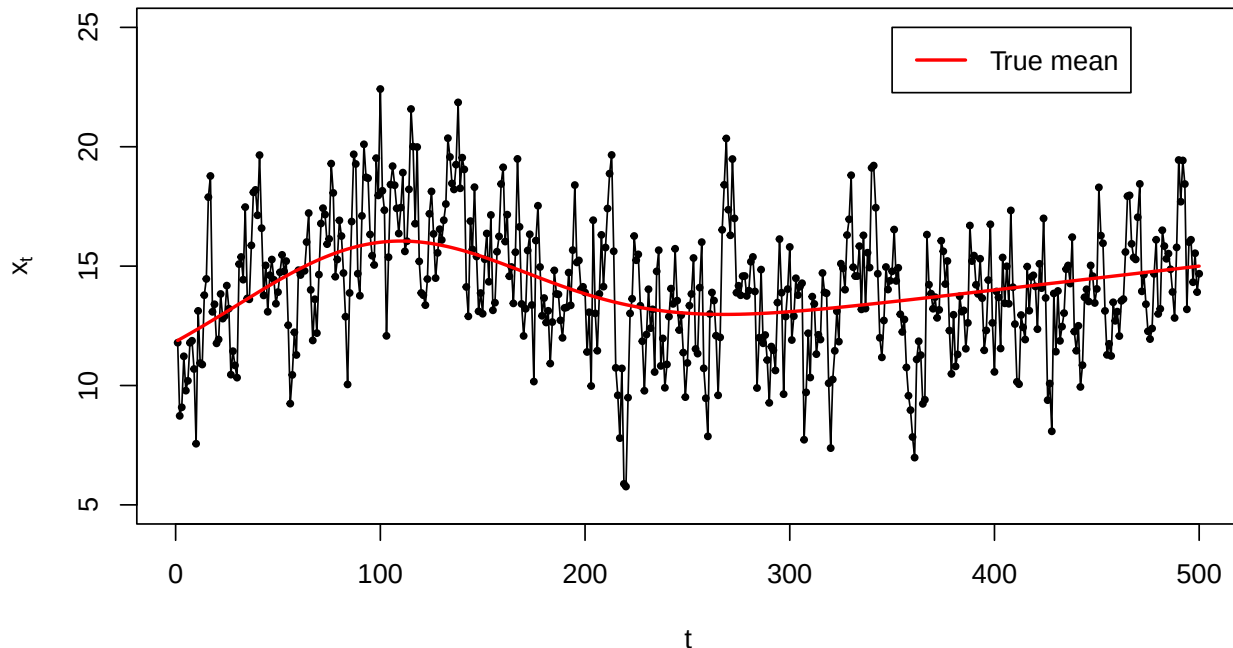
and $E_t \sim \text{AR}(1)$ with parameter $\phi_1 = 0.6$ based on a Gaussian IID process with variance $\sigma_Z^2 = 2^2$

```
set.seed(43)
n<-500
tv<-seq(1,n,by=1)
true.m<-10+0.01*tv+5*exp(-0.0001*(tv-100)^2)
phi<-0.6
sigZ<-2
et<-xt<-rep(0,n)
et[1]<-rnorm(1)*sigZ/sqrt(1-phi^2)
xt[1]<-true.m[1]+et[1]
```

```

for(i in 2:n){
  et[i]<-phi*et[i-1]+rnorm(1)*sigZ
  xt[i]<-true.m[i]+et[i]
}
plot(tv,xt,type='l',xlab='t',ylab=expression(x[t]),ylim=range(5,25))
points(tv,xt,pch=19,cex=0.5)
lines(tv,true.m,col='red',lwd=2)
legend(350,25,c('True mean'),lty=1,lwd=2,col='red')

```



We attempt smoothing by averaging with $q = 5$, and exponential smoothing with $\alpha = 0.1$.

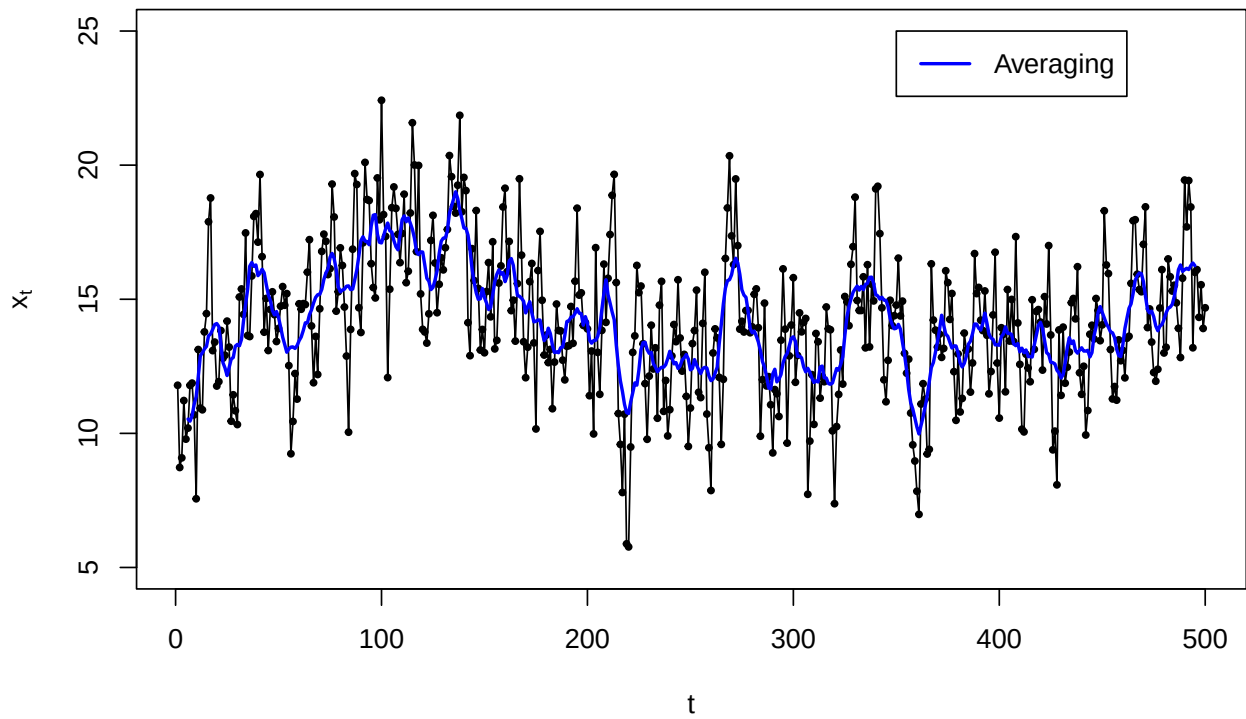
```

m1<-m2<-rep(0,n)

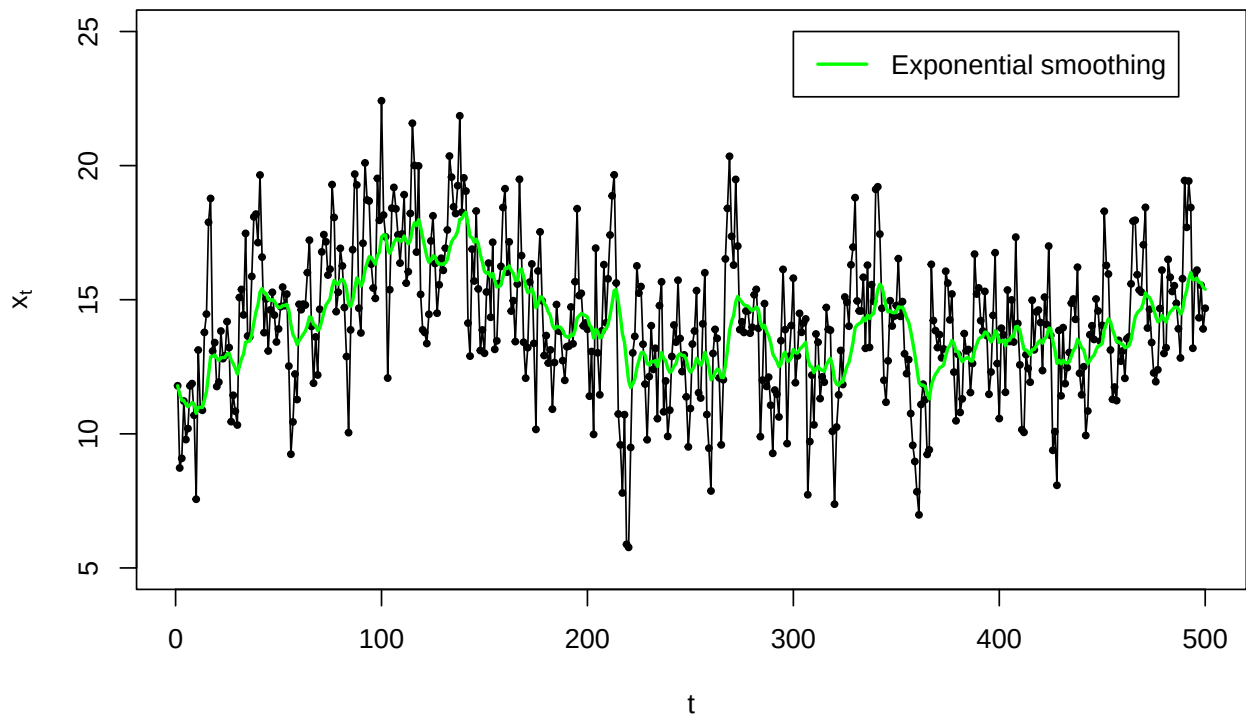
#Averaging
q<-5
for(i in (q+1):(n-q)){
  window<-(i-q):(i+q)
  m1[i]<-mean(xt[window])
}

#Exponential smoothing
alpha<-0.1
m2[1]<-xt[1]
for(i in 2:n){
  m2[i]<-alpha*xt[i]+(1-alpha)*m2[i-1]
}
par(mar=c(4,4,2,2))
plot(tv,xt,type='l',xlab='t',ylab=expression(x[t]),ylim=range(5,25))
points(tv,xt,pch=19,cex=0.5)
lines(tv[(q+1):(n-q)],m1[(q+1):(n-q)],col='blue',lwd=2)
legend(350,25,c('Averaging'),lty=1,lwd=2,col='blue')

```



```
plot(tv,xt,type='l',xlab='t',ylab=expression(x[t]),ylim=range(5,25))
points(tv,xt,pch=19,cex=0.5)
lines(tv,m2,col='green',lwd=2)
legend(300,25,c('Exponential smoothing'),lty=1,lwd=2,col='green')
```



General Strategy: For series $\{Y_t\}$

$$Y_t = m_t + s_t + X_t$$

where s_t has period d .

1. If d is even, set $q = d/2$, else set $q = (d - 1)/2$.
2. Use moving average for detrending respecting the seasonality

$$\hat{m}_t = \begin{cases} \frac{y_{t-q} + y_{t+q}}{2d} + \frac{1}{d} \sum_{j=-q+1}^{q-1} y_{t+j} & d \text{ even} \\ \frac{1}{d} \sum_{j=-q}^q y_{t+j} & d \text{ odd} \end{cases}$$

3. Remove seasonality: for $k = 1, \dots, d$, compute

$$w_k = \frac{1}{n_d} \sum_j (y_{k+jd} - \hat{m}_{k+jd})$$

where n_d is the number of terms in the sum (essentially the number of cycles in the series). For $k = 1, \dots, d$, set

$$\hat{s}_k = w_k - \frac{1}{d} \sum_{j=1}^d w_j$$

and set $\hat{s}_k = \hat{s}_{k-d}$ for $k = d + 1, d + 2, \dots, T$. The deseasonalized data are $y_t^* = y_t - \hat{s}_t$.

4. Recompute the deseasonalized trend $\hat{m}_t^*$ from the deseasonalized data y_t^* using the procedure in Step 2.
5. Compute the residual series

$$\hat{x}_t = y_t - \hat{m}_t^* - \hat{s}_t.$$

The resulting process $\hat{x}_t$ should be a zero mean, but potentially autocorrelated process.

EXAMPLE: Suppose we have the process

$$Y_t = m_t + s_t + X_t$$

for

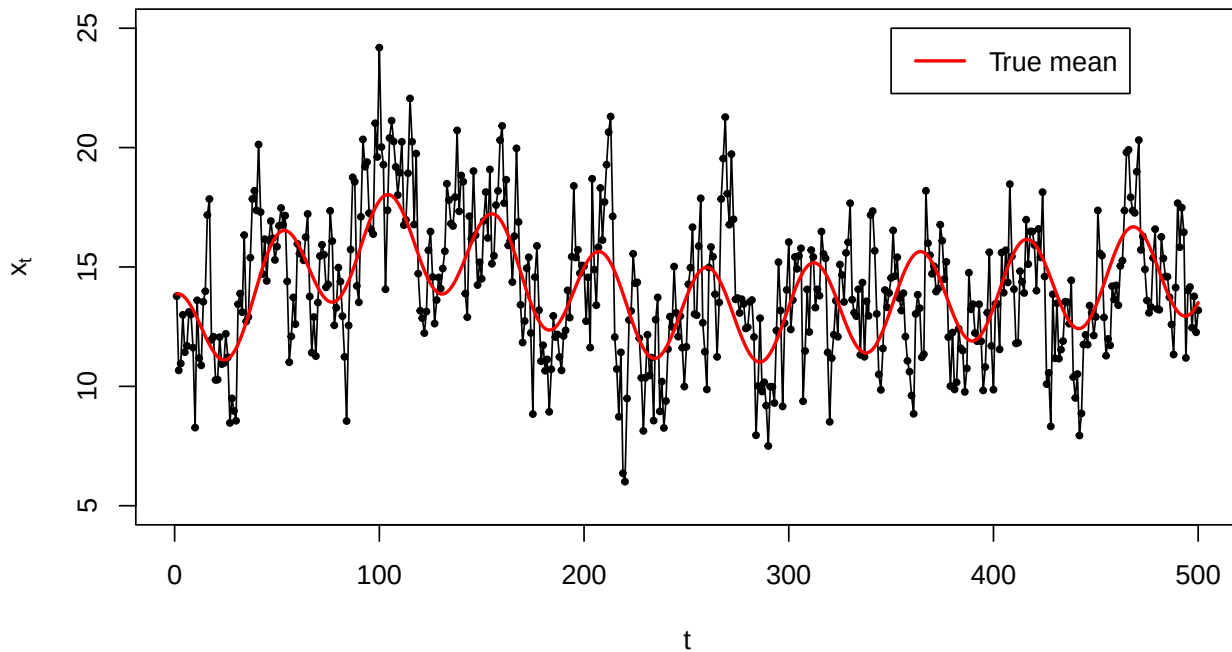
$$m_t = 10 + 0.01t + 5 \exp\{-0.001(t - 100)^2\}$$

and

$$s_t = 2 \cos(2\pi\lambda t)$$

so that $d = 1/\lambda$. In the simulation below, we take $\lambda = 1/52$, so $d = 52$. We again set $X_t \sim \text{AR}(1)$ with parameter $\phi_1 = 0.6$ based on a Gaussian IID process with variance $\sigma_Z^2 = 2^2$

```
set.seed(43)
n<-500
tv<-seq(1,n,by=1)
true.m<-10+0.01*tv+5*exp(-0.0001*(tv-100)^2)
lambda<-1/52
d<-1/lambda
true.s<-2*cos(2*pi*lambda*tv)
phi<-0.6
sigZ<-2
yt<-xt<-rep(0,n)
xt[1]<-rnorm(1)*sigZ/sqrt(1-phi^2)
yt[1]<-true.m[1]+true.s[1]+xt[1]
for(i in 2:n){
  xt[i]<-phi*xt[i-1]+rnorm(1)*sigZ
  yt[i]<-true.m[i]+true.s[i]+xt[i]
}
plot(tv,yt,type='l',xlab='t',ylab=expression(x[t]),ylim=range(5,25))
points(tv,yt,pch=19,cex=0.5)
lines(tv,true.m+true.s,col='red',lwd=2)
legend(350,25,c('True mean'),lty=1,lwd=2,col='red')
```

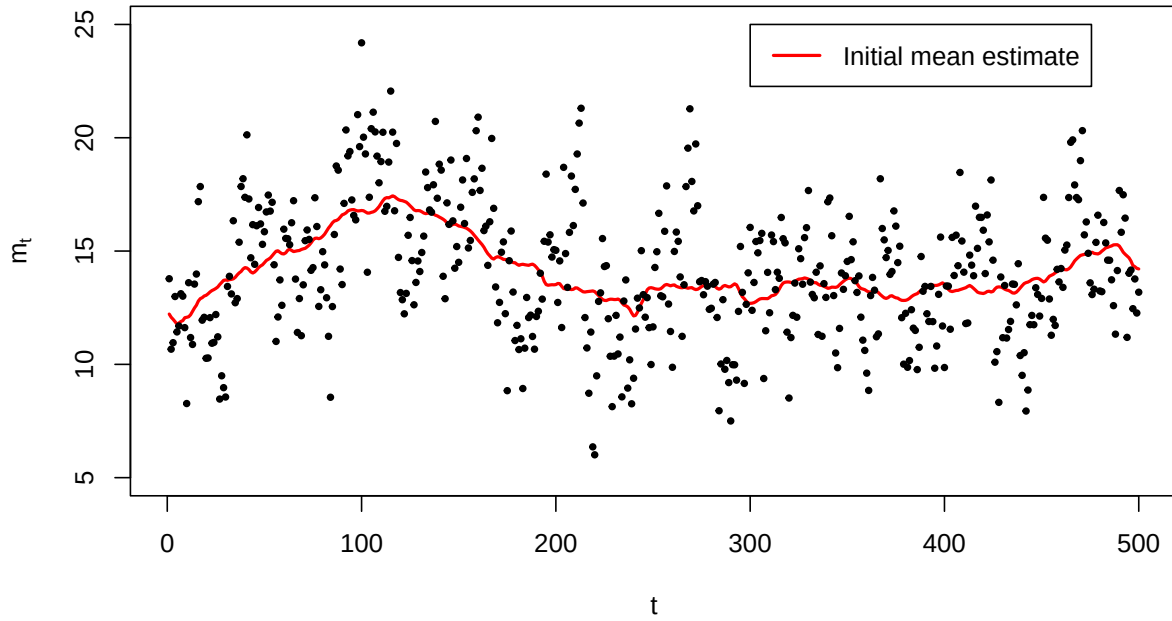


We implement the strategy as follows:

1. Here $d = 52$, so we set $q = d/2 = 26$.
2. For each t , we use the averaging formula

$$\hat{m}_t = \frac{y_{t-q} + y_{t+q}}{2d} + \frac{1}{d} \sum_{j=-q+1}^{q-1} y_{t+j}$$

```
q<-d/2
mt<-rep(0,n)
for(i in 1:q){
  window<-(i-q+1):(i+q-1)
  window<-window[window>0]
  mt[i]<-mean(yt[window])
}
for(i in (q+1):(n-q)){
  mt[i]<-sum(yt[(i-q+1):(i+q-1)])/d+(yt[i-q]+yt[i+q])/(2*d)
}
for(i in (n-q+1):n){
  window<-(i-q+1):(i+q-1)
  window<-window[window<=n]
  mt[i]<-mean(yt[window])
}
plot(tv,mt,type='l',xlab='t',lwd=2,ylab=expression(m[t]),ylim=range(5,25),col='red')
points(tv,yt,pch=19,cex=0.5)
legend(300,25,c('Initial mean estimate'),lty=1,lwd=2,col='red')
```



3. For $k = 1, \dots, d$, compute

$$w_k = \frac{1}{n_d} \sum_j (y_{k+jd} - \hat{m}_{k+jd})$$

where n_d is the number of terms in the sum (essentially the number of cycles in the series). For $k = 1, \dots, d$, set

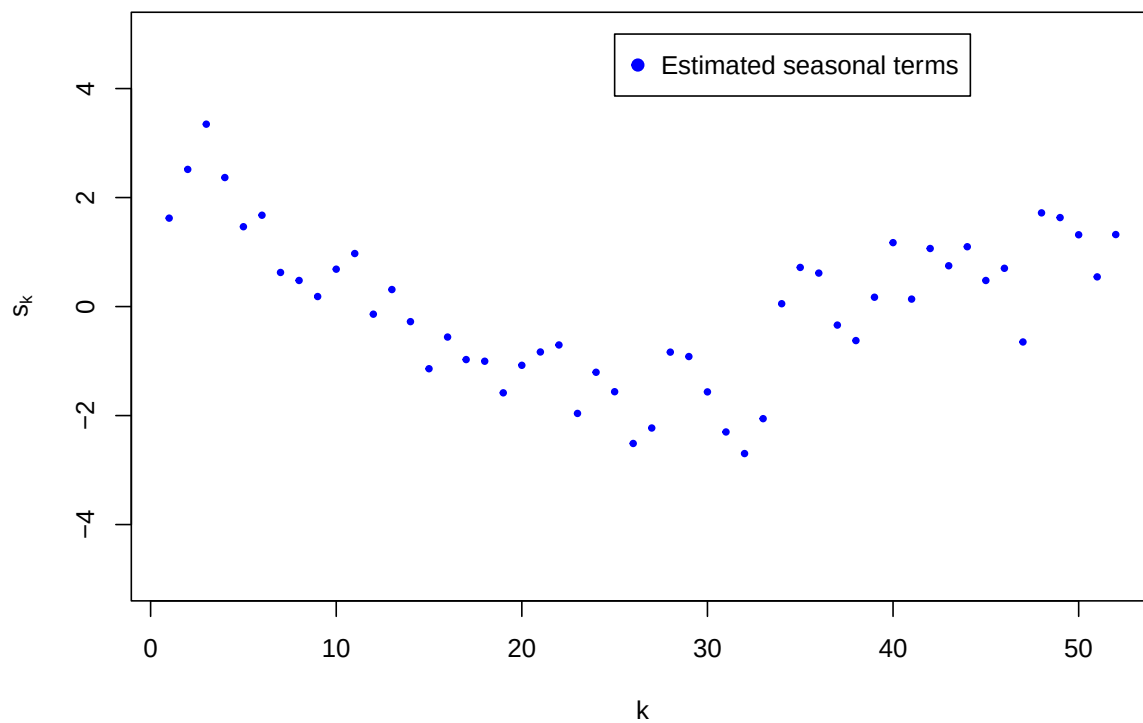
$$\hat{s}_k = w_k - \frac{1}{d} \sum_{j=1}^d w_j$$

and set $\hat{s}_k = \hat{s}_{k-d}$ for $k = d+1, d+2, \dots, T$. The deseasonalized data are $y_t^* = y_t - \hat{s}_t$.

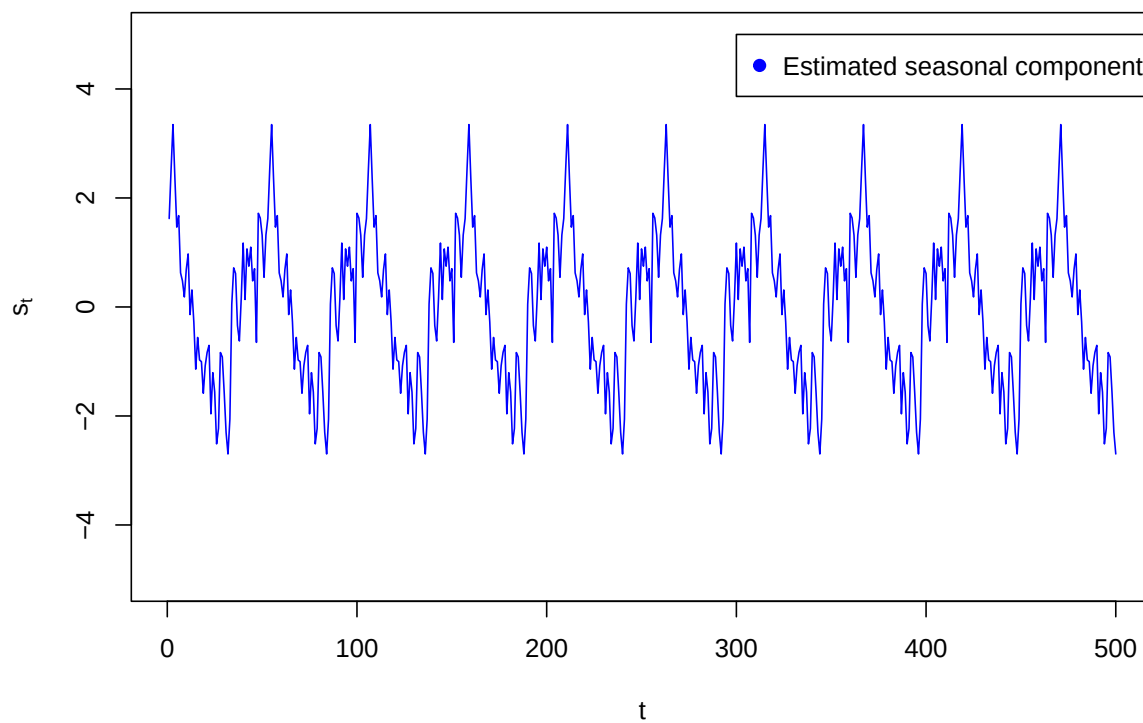
```

wk<-rep(0,d)
for(k in 1:d){
  indices<-k+c(1:11)*d
  indices<-indices[indices<=n]
  wk[k]<-mean(yt[indices]-mt[indices])
}
sk<-wk-mean(wk)
par(mar=c(4,5,2,2))
plot(1:d,sk,xlab='k',pch=19,cex=0.5,ylab=expression(s[k]),ylim=range(-5,5),col='blue')
legend(25,5,c('Estimated seasonal terms'),pch=19,col='blue')

```



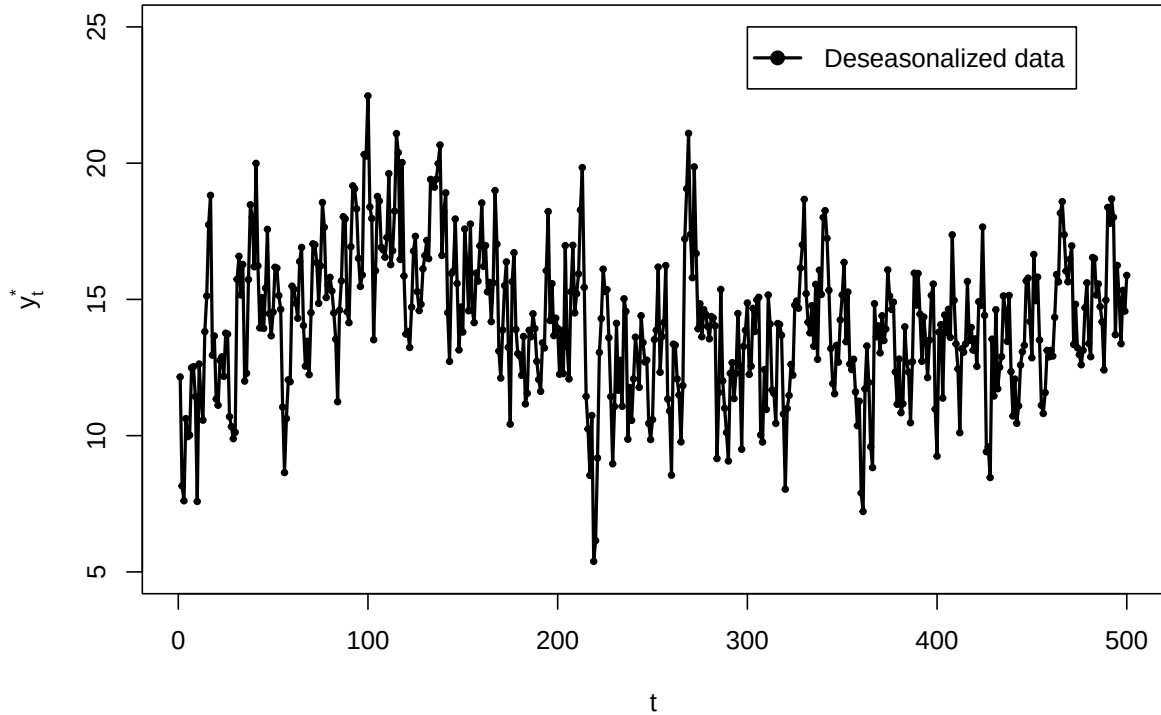
```
st<-rep(sk,11)
st<-st[1:n]
plot(tv,st,xlab='t',pch=19,cex=0.5,type='l',ylab=expression(s[t]),ylim=range(-5,5),col='blue')
legend(300,5,c('Estimated seasonal component'),pch=19,col='blue')
```



```

ystar<-yt-st
plot(tv,ystar,type='l',xlab='t',lwd=2,ylab=expression(paste(y[t]^"*")),ylim=range(5,25))
points(tv,ystar,pch=19,cex=0.5)
legend(300,25,c('Deseasonalized data'),lty=1,lwd=2,pch=19)

```

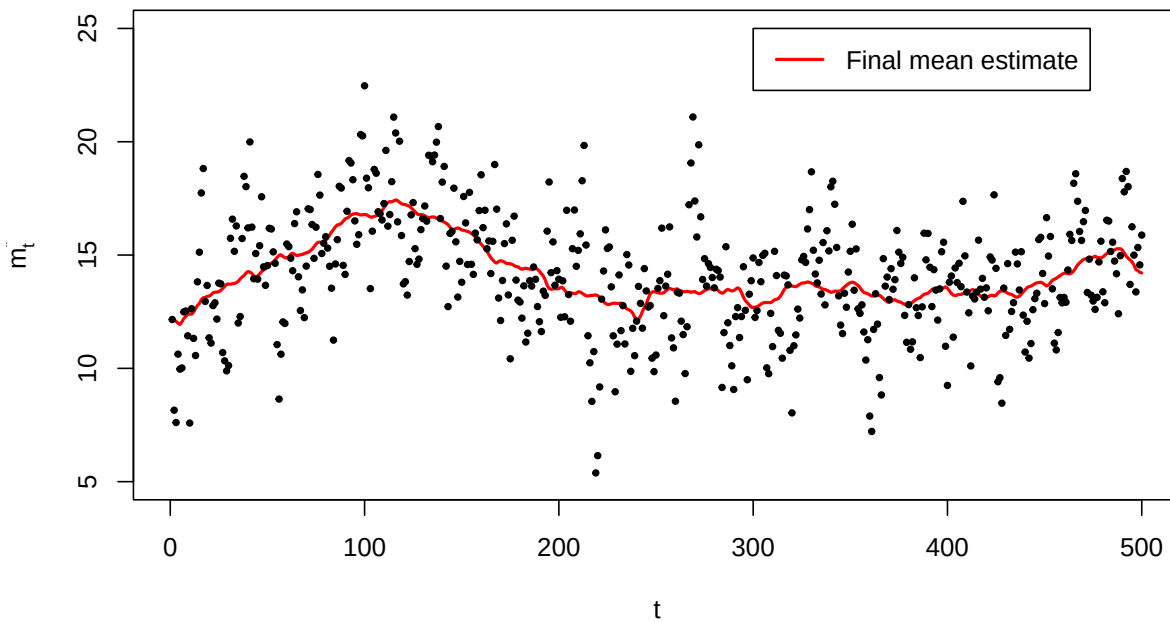


4. Recompute the deseasonalized trend $\hat{m}_t^*$ from the deseasonalized data y_t^* using the procedure in Step 2.

```

mstar<-rep(0,n)
for(i in 1:q){
  window<-(i-q+1):(i+q-1)
  window<-window[window>0]
  mstar[i]<-mean(ystar[window])
}
for(i in (q+1):(n-q)){
  mstar[i]<-sum(ystar[(i-q+1):(i+q-1)])/d+(ystar[i-q]+ystar[i+q])/(2*d)
}
for(i in (n-q+1):n){
  window<-(i-q+1):(i+q-1)
  window<-window[window<=n]
  mstar[i]<-mean(yt[window])
}
plot(tv,mstar,type='l',xlab='t',lwd=2,ylab=expression(paste(m[t]^"*")),
     ylim=range(5,25),col='red')
points(tv,ystar,pch=19,cex=0.5)
legend(300,25,c('Final mean estimate'),lty=1,lwd=2,col='red')

```

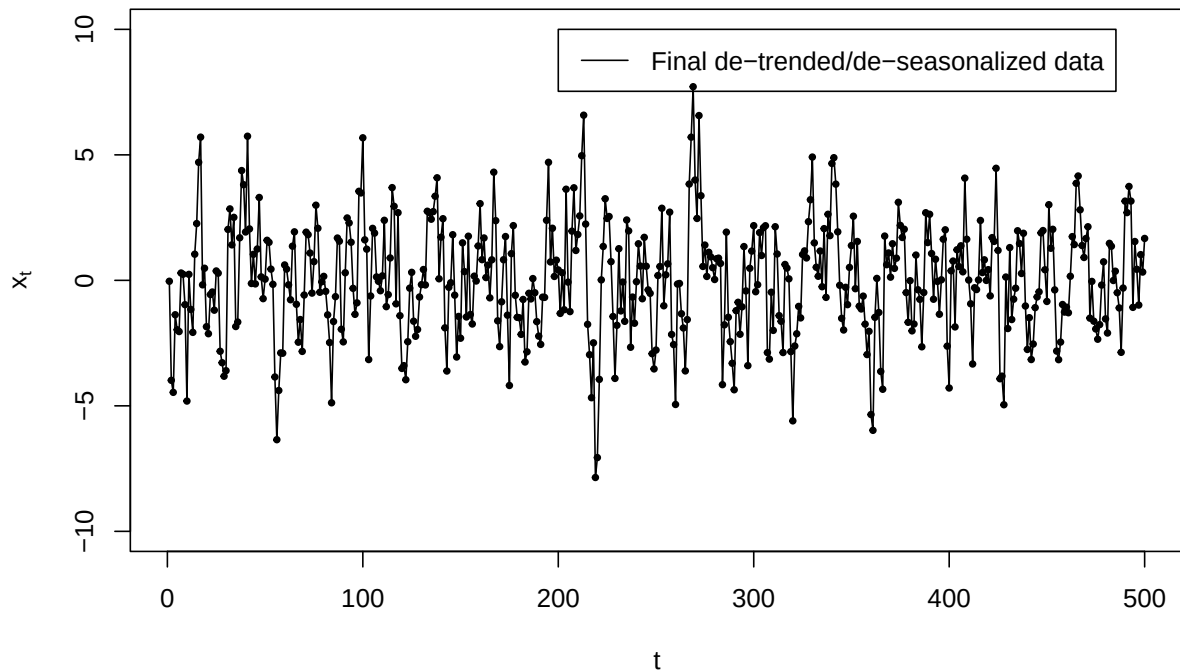


5. Compute the residual series

$$\hat{x}_t = y_t - \hat{m}_t^* - \hat{s}_t.$$

We can examine the properties of this residual series

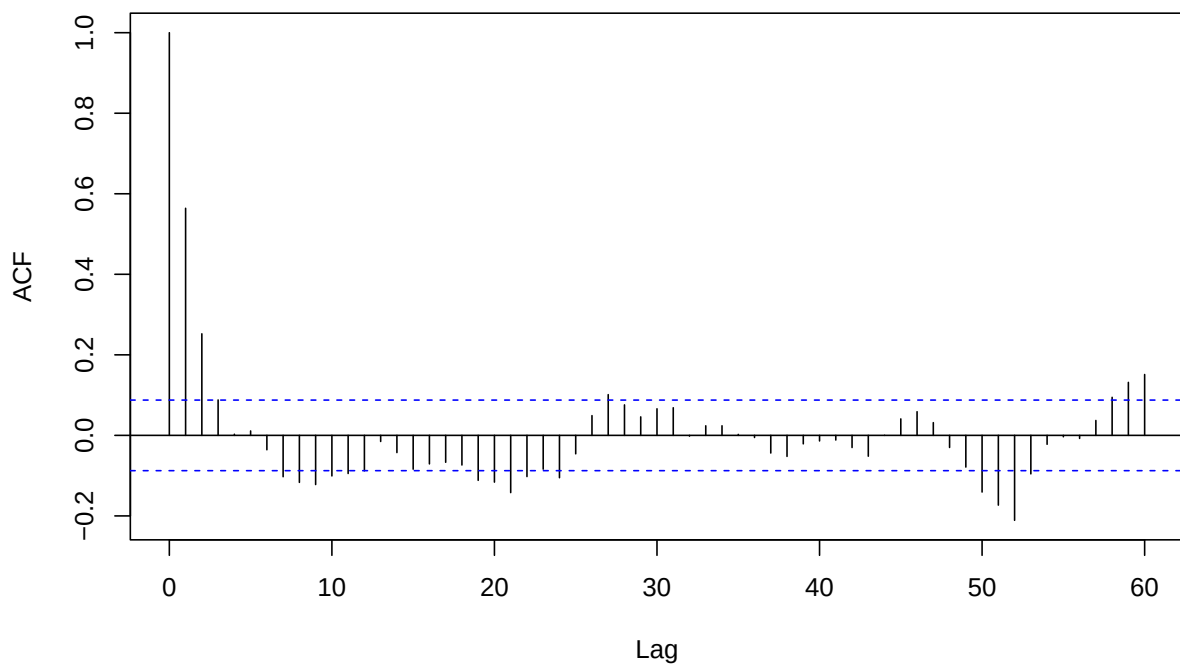
```
xf<-yt-mstar-st
par(mar=c(4,4,1,2))
plot(tv,xf,type='l',xlab='t',ylab=expression(x[t]),ylim=range(-10,10))
points(tv,xf,pch=19,cex=0.5);
legend(200,10,c('Final de-trended/de-seasonalized data'),lty=1)
```



The residual series seems to be zero mean and constant variance, and the estimated ACF sequence resembles that of a AR(1) process, although there is still some apparent excess correlation around lag $d = 52$.

```
par(mar=c(4,4,4,2))
acf(xf,lag.max=60,main='ACF of residual series')
```

ACF of residual series



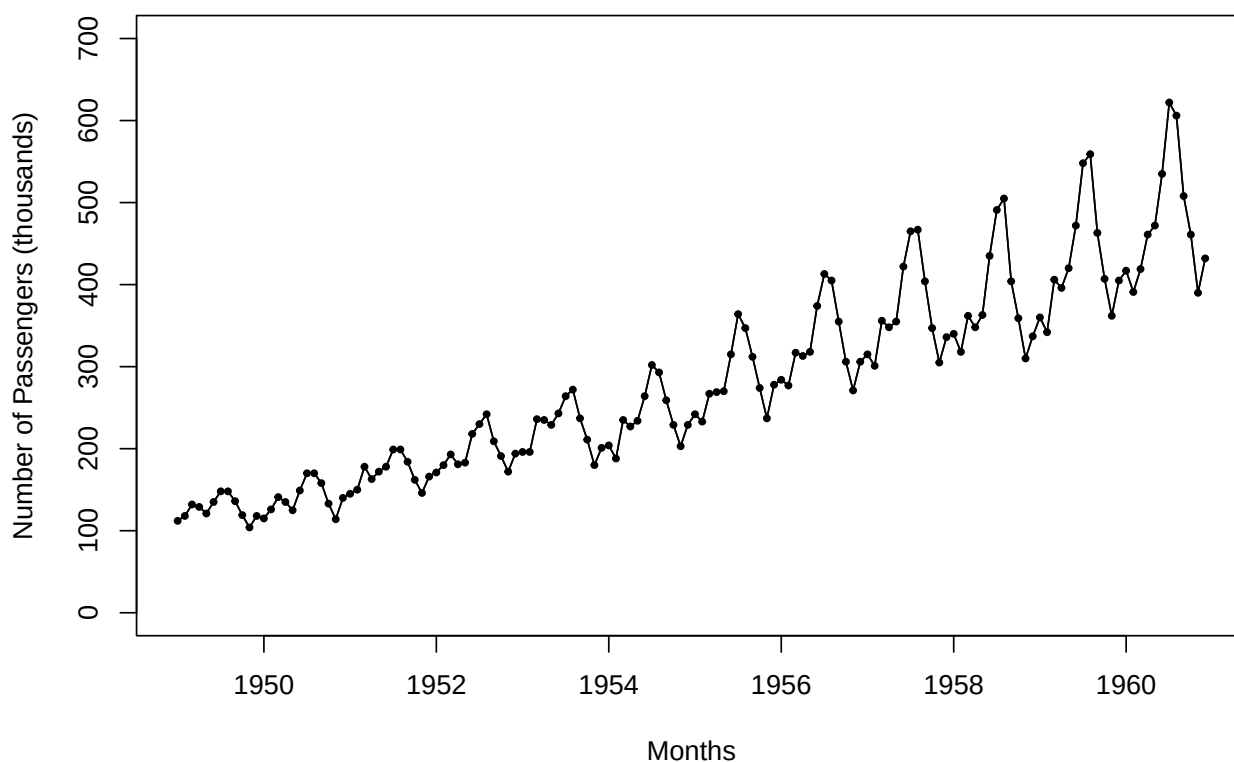
Real data example: Air Passenger data

These data represent the counts of the number (in thousands) of air passengers in the US between January 1949 and December 1960.

```
Air<-read.csv('https://www.math.mcgill.ca/dstephens/545/Data/airline-passengers.csv')
head(Air,n=4L)

##      Month Passengers
## 1 1949-01         112
## 2 1949-02         118
## 3 1949-03         132
## 4 1949-04         129

AirPassengers<-Air[,2]
y<-ts(AirPassengers,start=c(1949,1),frequency=12)
n<-length(y)
par(mar=c(4,4,2,2))
plot(y,type="l",ylim=range(0,700),ylab="Number of Passengers (thousands)",xlab="Months")
lines(y)
points(y,pch=19,cex=0.5)
legend(0,600,c("Data"),lty=c(1,1),col=c("black"),pch=c(19))
```



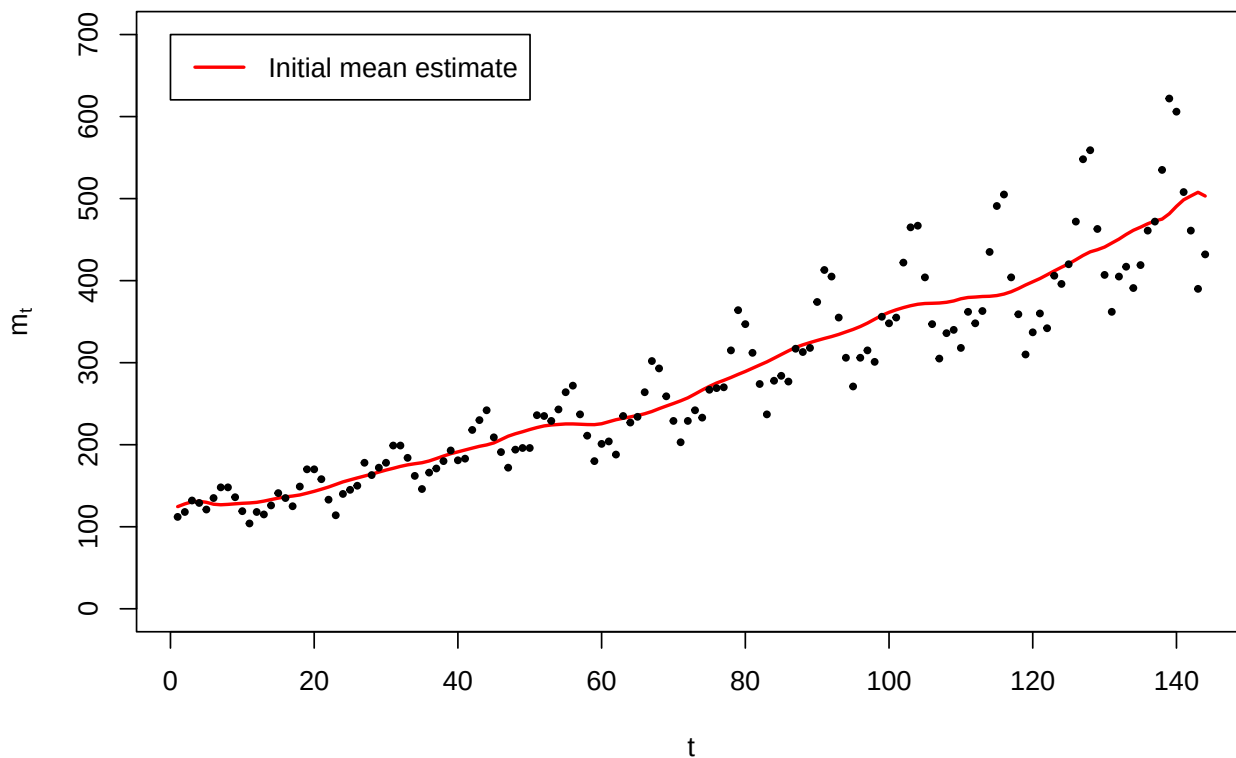
The data are recorded monthly, and there is an apparent annual cycle. We follow the procedure above, with $d = 12$ (and hence $\lambda = 1/12$).

```
yt<-as.numeric(Air[,2])
tv<-1:n
d<-12
q<-d/2
mt<-rep(0,n)
for(i in 1:q){
  window<-(i-q+1):(i+q-1)
```

```

    window<-window[window>0]
    mt[i]<-mean(yt[window])
  }
  for(i in (q+1):(n-q)){
    mt[i]<-sum(yt[(i-q+1):(i+q-1)])/(d+(yt[i-q]+yt[i+q])/(2*d))
  }
  for(i in (n-q+1):n){
    window<-(i-q+1):(i+q-1)
    window<-window[window<=n]
    mt[i]<-mean(yt[window])
  }
  par(mar=c(4,4,2,2))
  plot(tv,mt,type='l',xlab='t',lwd=2,ylab=expression(m[t]),ylim=range(0,700),col='red')
  points(tv,yt,pch=19,cex=0.5)
  legend(0,700,c('Initial mean estimate'),lty=1,lwd=2,col='red')

```



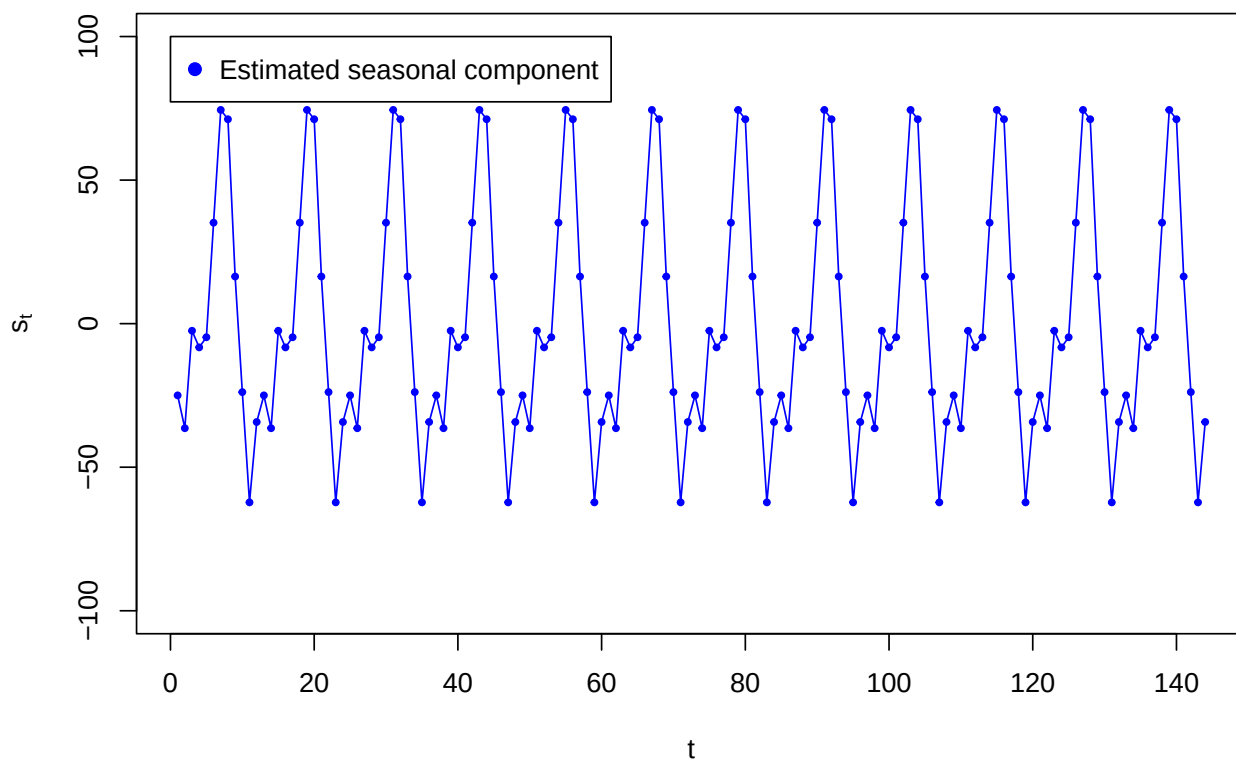
```

wk<-rep(0,d)
for(k in 1:d){
  indices<-k+c(1:21)*d
  indices<-indices[indices<=n]
  wk[k]<-mean(yt[indices]-mt[indices])
}
sk<-wk-mean(wk)
sk #Estimated seasonal components

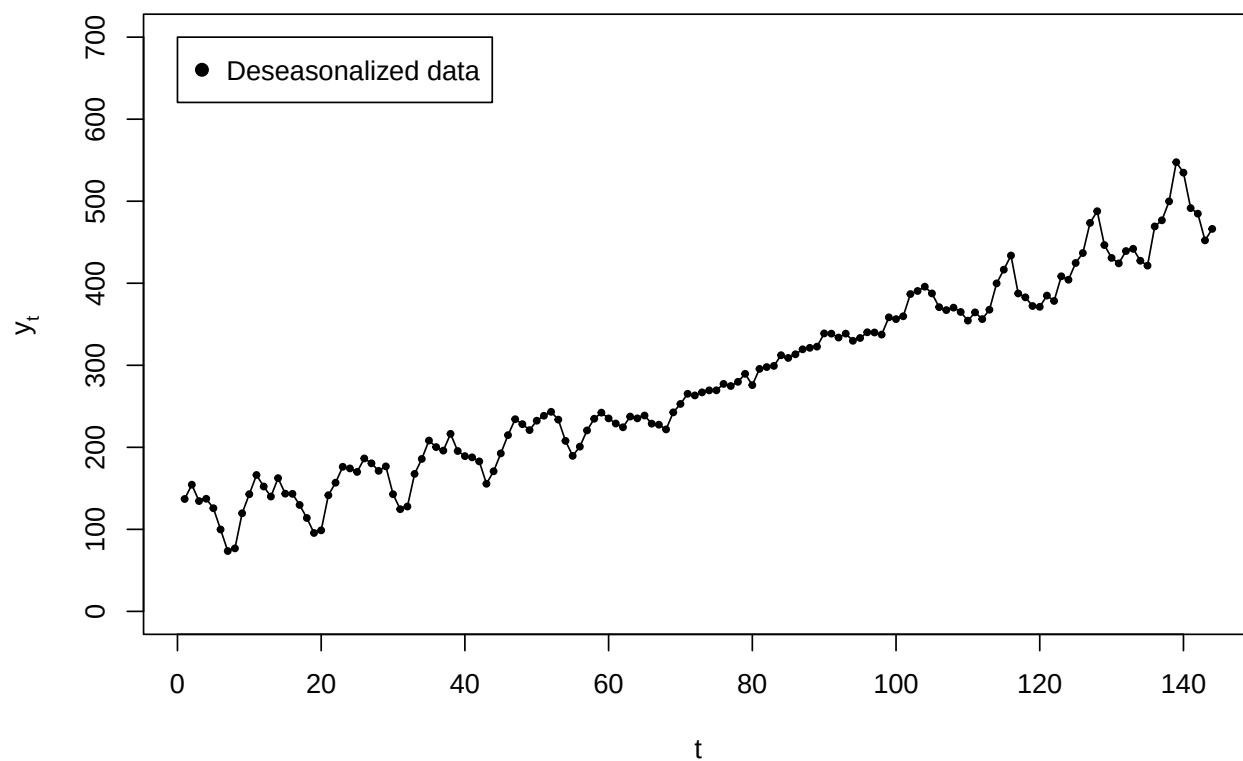
## [1] -24.979808 -36.419202 -2.472233 -8.267687 -4.737384 35.171707
## [7] 74.440302 71.196707 16.416656 -23.843445 -62.253077 -34.252536

st<-rep(sk,21)
st<-st[1:n]; ystar<-yt-st
plot(tv,st,type='l',xlab='t',pch=19,cex=0.5,ylab=expression(s[t]),ylim=range(-100,100),col='blue')
points(tv,yt,pch=19,cex=0.5,col='blue')
legend(0,100,c('Estimated seasonal component'),pch=19,col='blue')

```



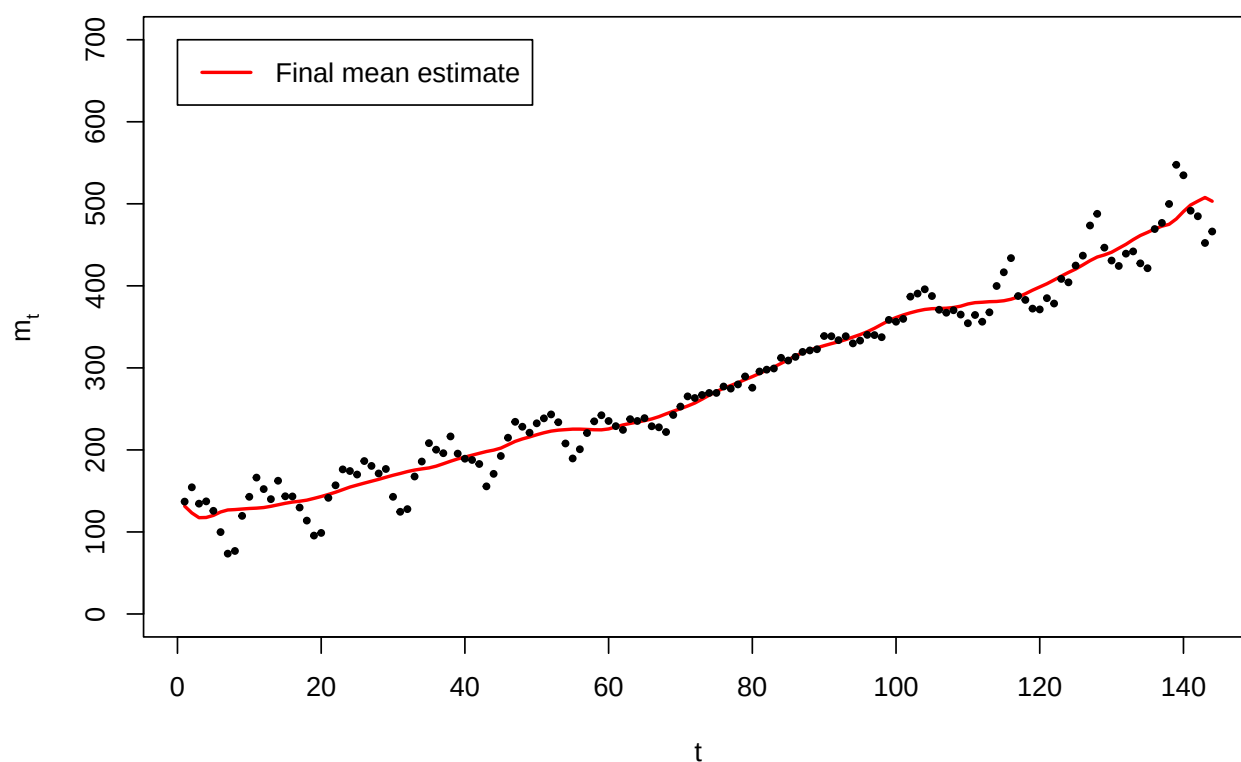
```
plot(tv,ystar,type='l',xlab='t',pch=19,cex=0.5,ylab=expression(paste(y[t]^"*")),ylim=range(0,700))
points(tv,ystar,pch=19,cex=0.5)
legend(0,700,c('Deseasonalized data'),pch=19)
```



```

mstar<-rep(0,n)
for(i in 1:q){
  window<-(i-q+1):(i+q-1); window<-window[window>0]
  mstar[i]<-mean(ystar[window])
}
for(i in (q+1):(n-q)){
  mstar[i]<-sum(ystar[(i-q+1):(i+q-1)])/d+(ystar[i-q]+ystar[i+q])/(2*d)
}
for(i in (n-q+1):n){
  window<-(i-q+1):(i+q-1); window<-window[window<=n]
  mstar[i]<-mean(yt[window])
}
plot(tv,mstar,type='l',xlab='t',lwd=2,ylab=expression(paste(m[t]^"*")),
      ylim=range(0,700),col='red')
points(tv,ystar,pch=19,cex=0.5)
legend(0,700,c('Final mean estimate'),lty=1,lwd=2,col='red')

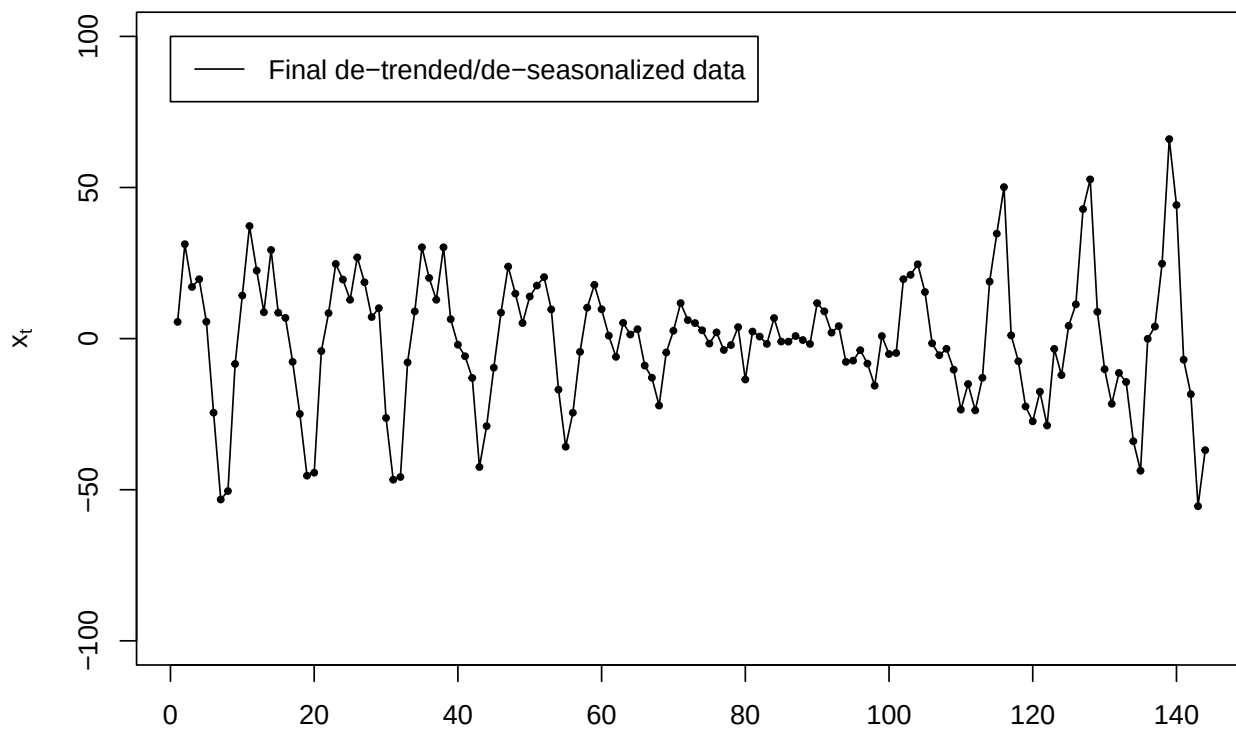
```



```

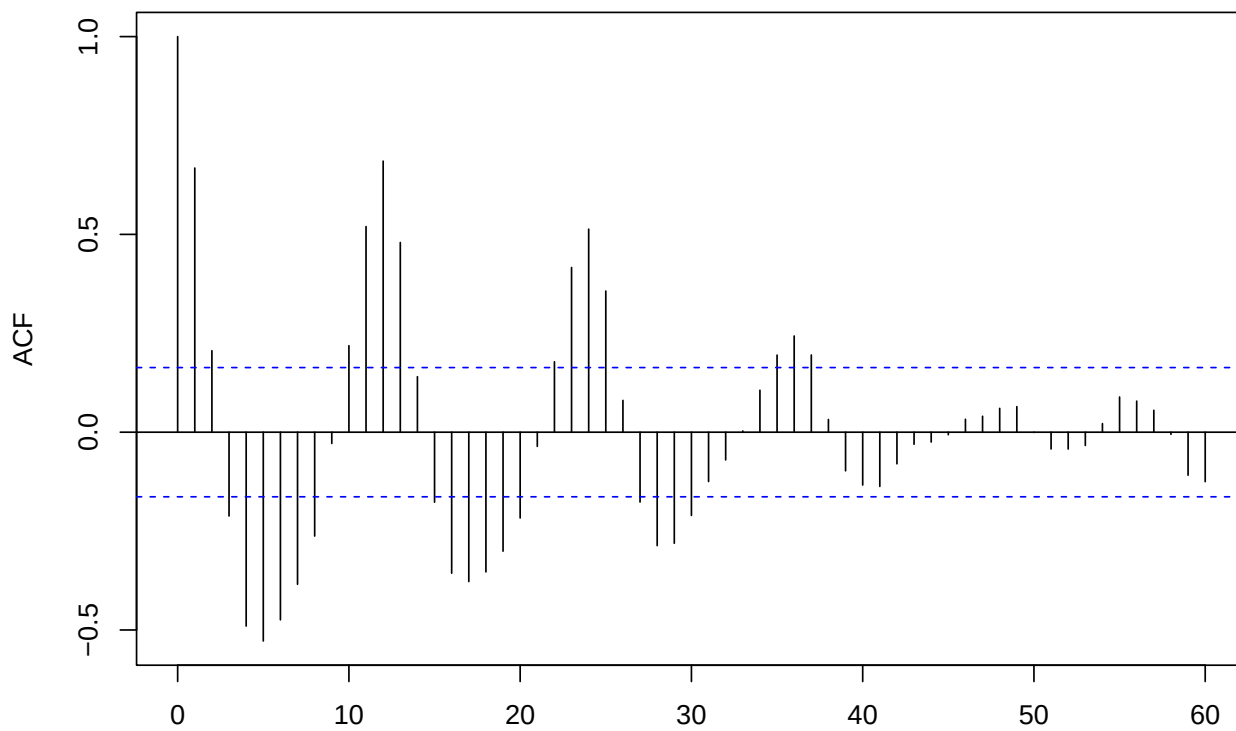
xf<-yt-mstar-st
par(mar=c(2,4,3,2))
plot(tv,xf,type='l',xlab='t',ylab=expression(x[t]),ylim=range(-100,100))
points(tv,xf,pch=19,cex=0.5)
legend(0,100,c('Final de-trended/de-seasonalized data'),lty=1)

```



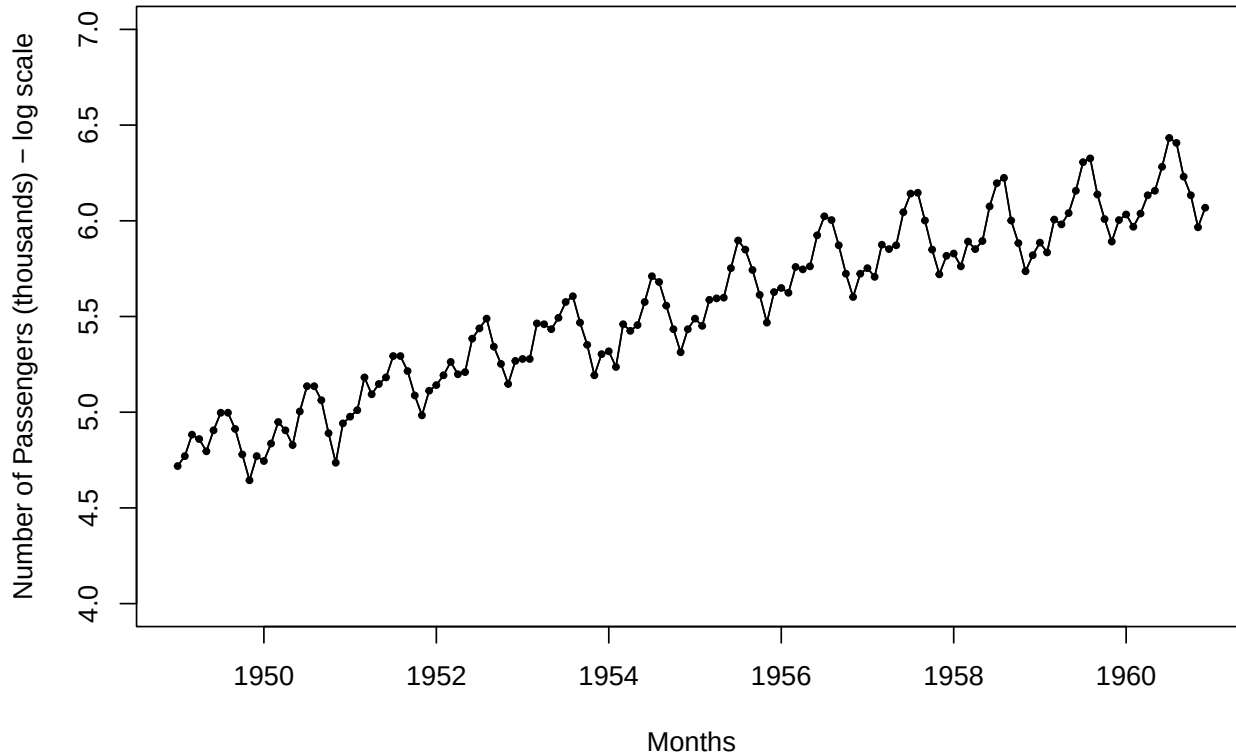
```
acf(xf,lag.max=60,main='ACF of residual series')
```

ACF of residual series



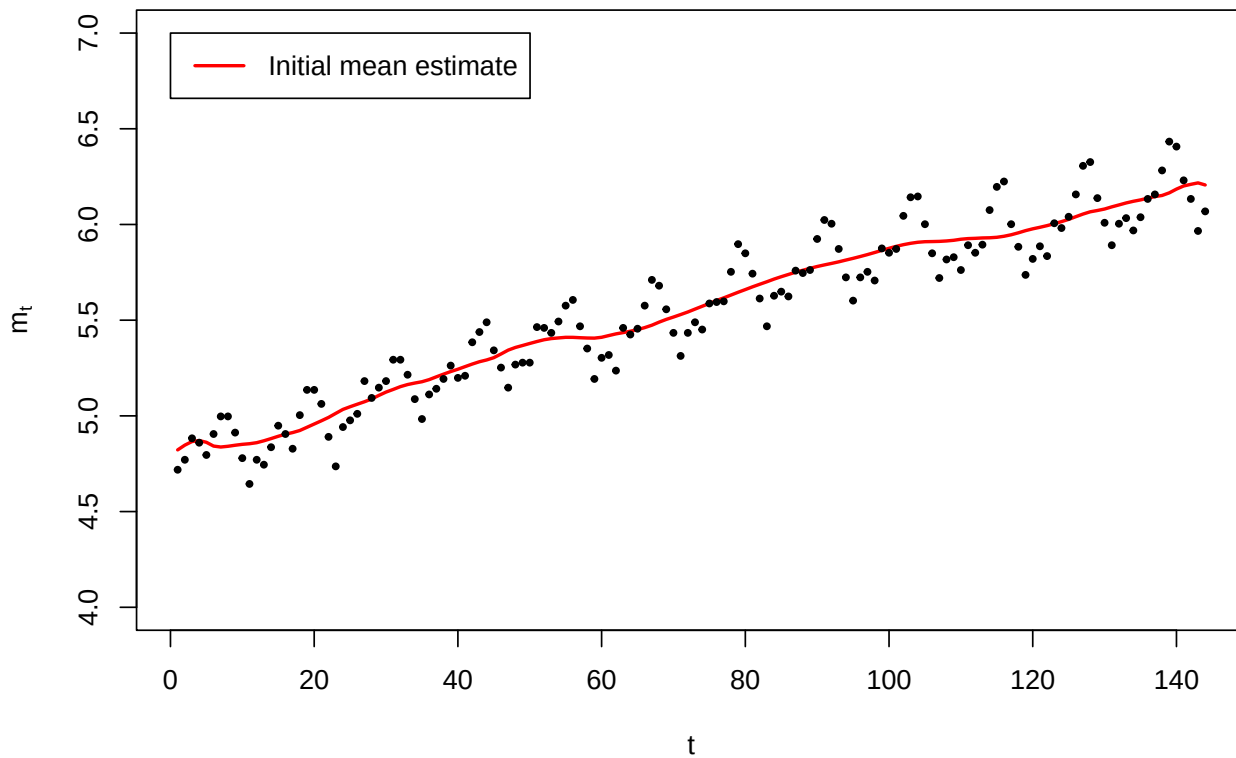
We can see here that the seasonality has not been removed successfully. This is because the amplitude of the seasonal variation is not constant. This can be partly addressed by taking a log transform of the data.

```
y<-ts(log(AirPassengers),start=c(1949,1),frequency=12)
n<-length(y)
par(mar=c(4,4,2,2))
plot(y,type="l",ylim=range(4,7),ylab="Number of Passengers (thousands) - log scale",xlab="Months")
lines(y)
points(y,pch=19,cex=0.5)
legend(0,7,c("Data"),lty=c(1,1),col=c("black"),pch=c(19))
```



```
yt<-as.numeric(log(Air[,2]))
tv<-1:n
d<-12
q<-d/2
mt<-rep(0,n)
for(i in 1:q){
  window<-(i-q+1):(i+q-1)
  window<-window[window>0]
  mt[i]<-mean(yt[window])
}
for(i in (q+1):(n-q)){
  mt[i]<-sum(yt[(i-q+1):(i+q-1)])/d+(yt[i-q]+yt[i+q])/(2*d)
}
for(i in (n-q+1):n){
  window<-(i-q+1):(i+q-1)
  window<-window[window<=n]
  mt[i]<-mean(yt[window])
}
par(mar=c(4,4,2,2))
plot(tv,mt,type='l',xlab='t',lwd=2,ylab=expression(m[t]),ylim=range(4,7),col='red')
```

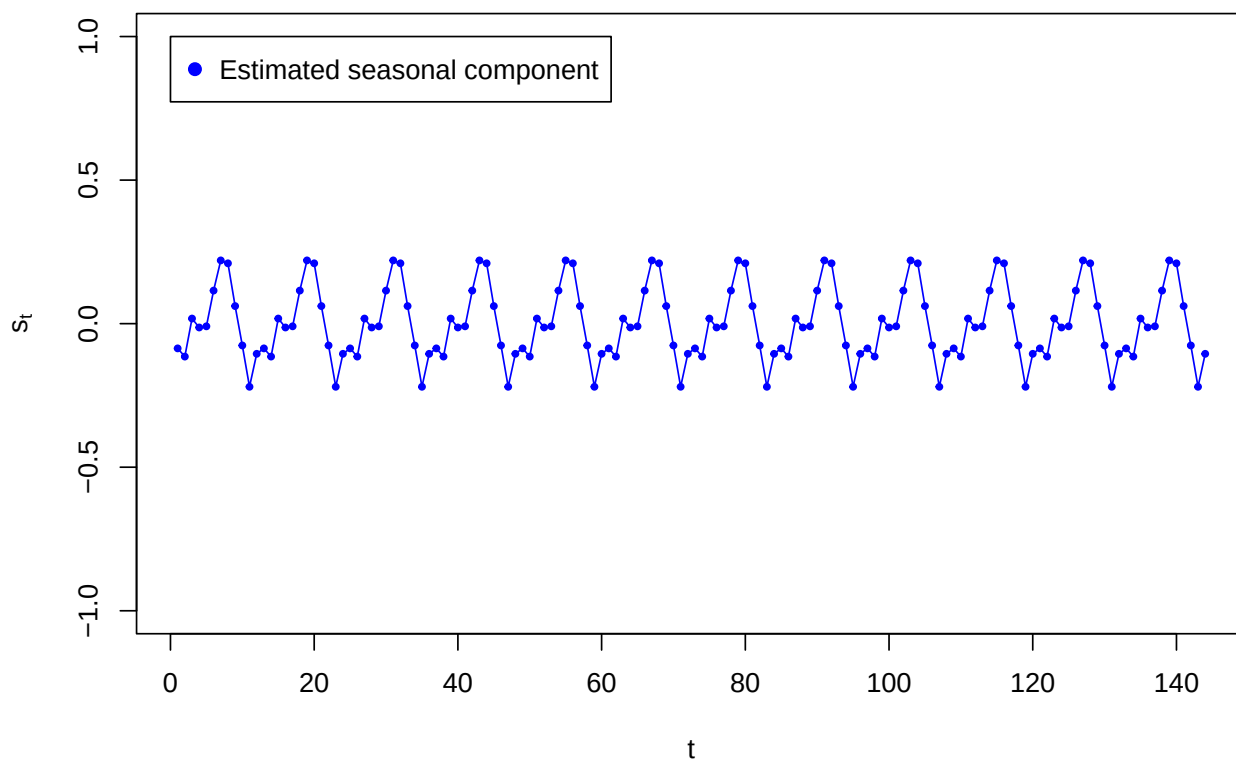
```
points(tv,yt,pch=19,cex=0.5)
legend(0,7,c('Initial mean estimate'),lty=1,lwd=2,col='red')
```



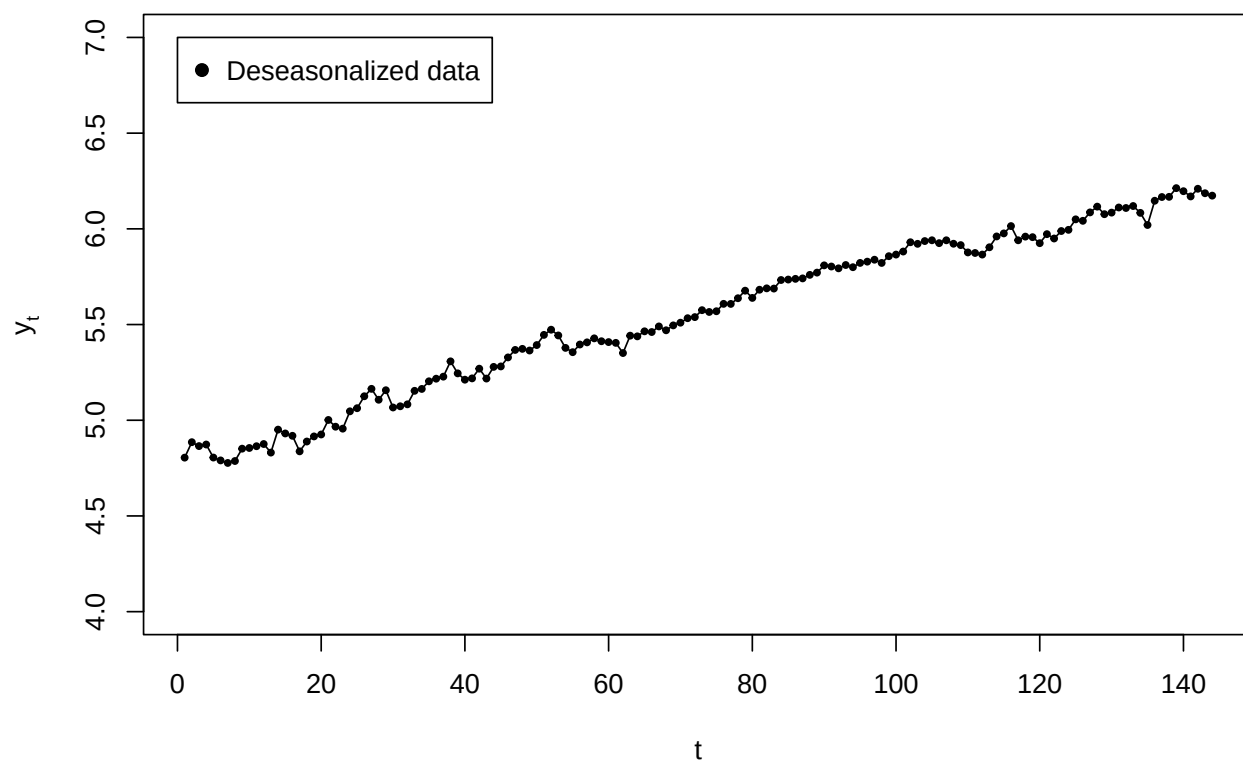
```
wk<-rep(0,d)
for(k in 1:d){
  indices<-k+c(1:21)*d
  indices<-indices[indices<=n]
  wk[k]<-mean(yt[indices]-mt[indices])
}
sk<-wk-mean(wk)
sk #Estimated seasonal components

## [1] -0.086141215 -0.114739044 0.017787159 -0.013371807 -0.009292302
## [6] 0.115066801 0.220288505 0.210177431 0.061165577 -0.075977162
## [11] -0.219896910 -0.105067034

st<-rep(sk,21)
st<-st[1:n]
plot(tv,st,type='l',xlab='t',pch=19,cex=0.5,ylab=expression(s[t]),ylim=range(-1,1),col='blue')
points(tv,st,pch=19,cex=0.5,col='blue')
legend(0,1,c('Estimated seasonal component'),pch=19,col='blue')
```



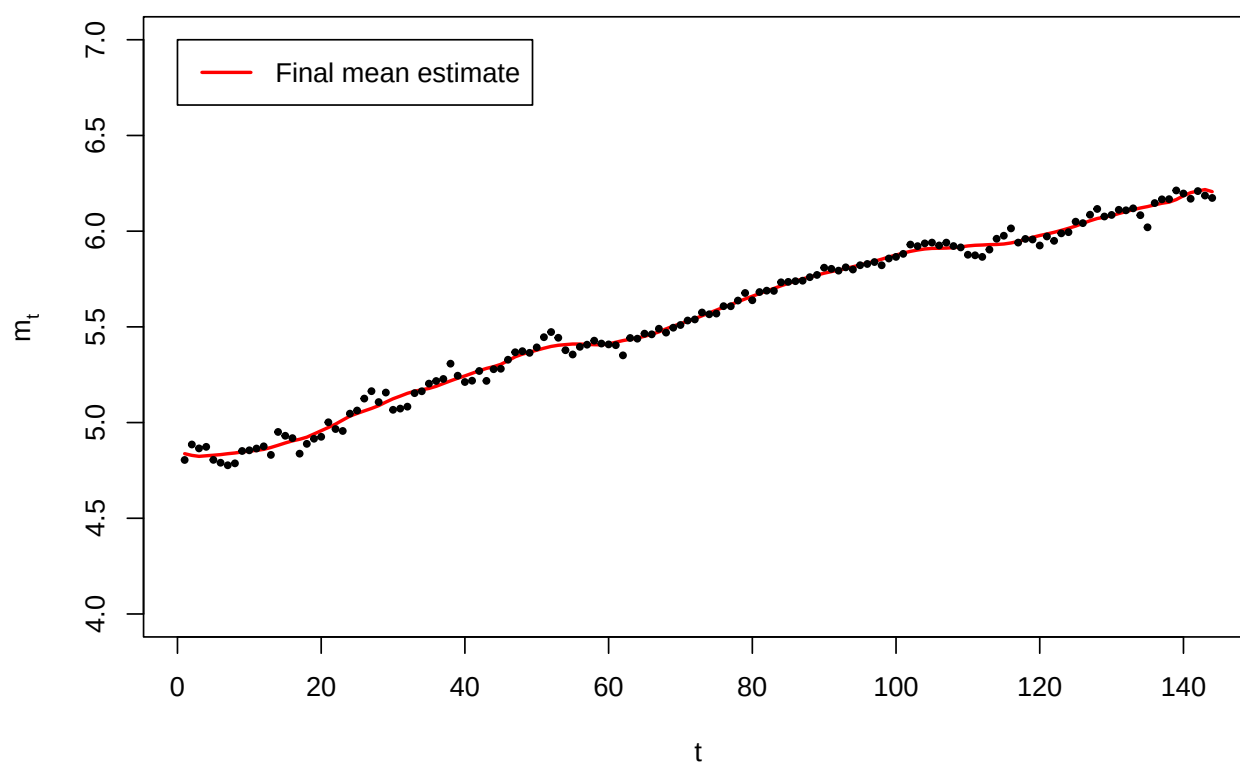
```
ystar<-yt-st
plot(tv,ystar,type='l',xlab='t',pch=19,cex=0.5,ylab=expression(paste(y[t]^"*")),ylim=range(4,7))
points(tv,ystar,pch=19,cex=0.5); legend(0,7,c('Deseasonalized data'),pch=19)
```



```

mstar<-rep(0,n)
for(i in 1:q){
  window<-(i-q+1):(i+q-1); window<-window[window>0]
  mstar[i]<-mean(ystar[window])
}
for(i in (q+1):(n-q)){
  mstar[i]<-sum(ystar[(i-q+1):(i+q-1)])/d+(ystar[i-q]+ystar[i+q])/(2*d)
}
for(i in (n-q+1):n){
  window<-(i-q+1):(i+q-1); window<-window[window<=n]
  mstar[i]<-mean(yt[window])
}
plot(tv,mstar,type='l',xlab='t',lwd=2,ylab=expression(paste(m[t]^"*")),
      ylim=range(4,7),col='red')
points(tv,ystar,pch=19,cex=0.5)
legend(0,7,c('Final mean estimate'),lty=1,lwd=2,col='red')

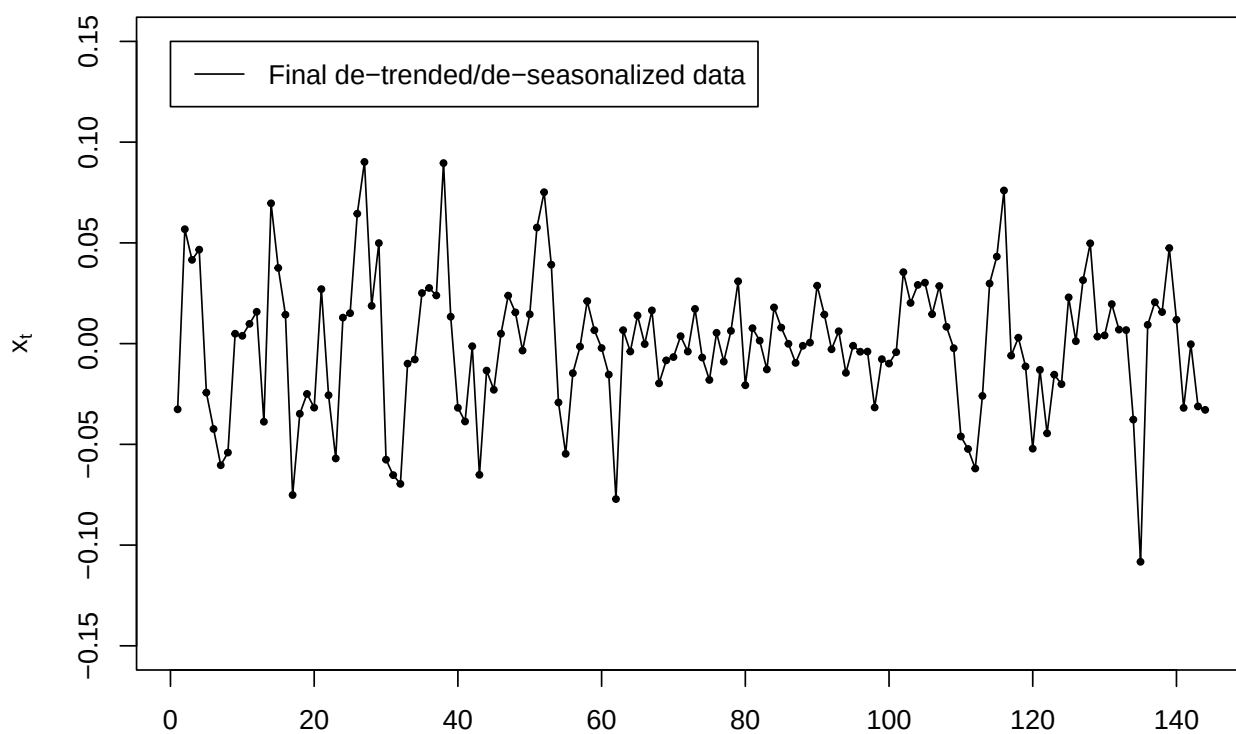
```



```

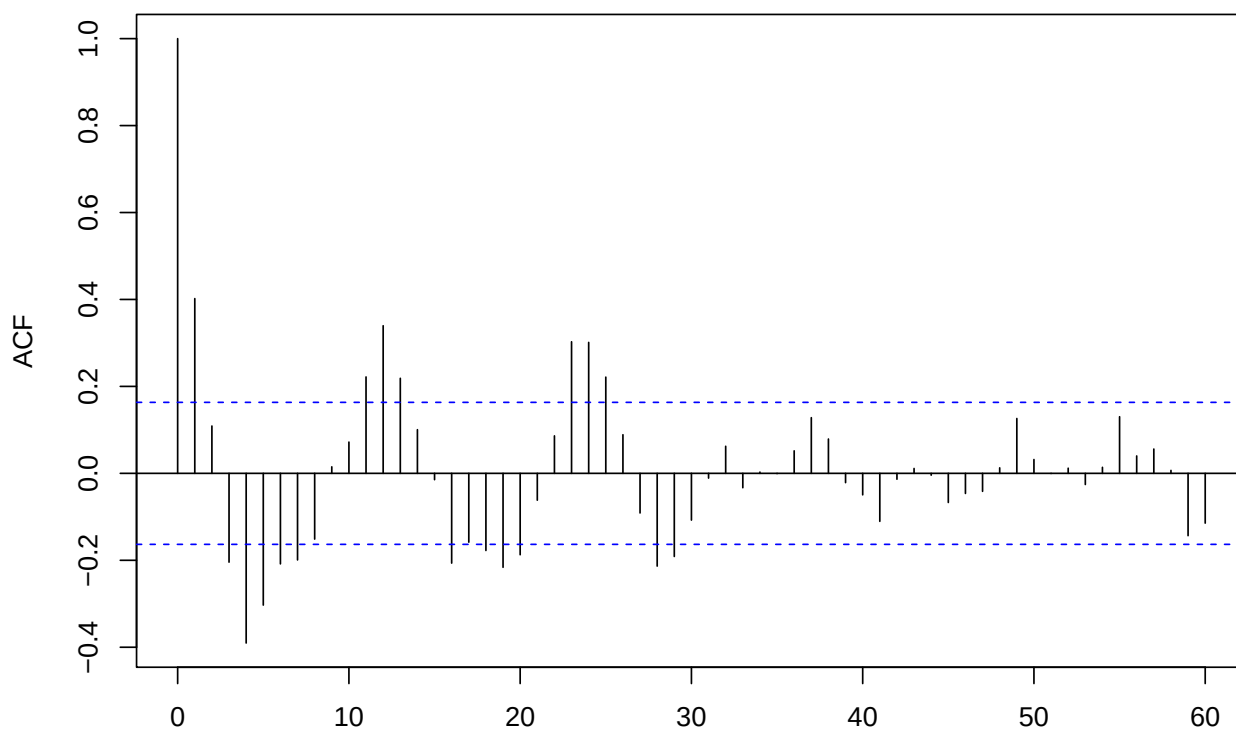
xf<-yt-mstar-st
par(mar=c(2,4,3,2))
plot(tv,xf,type='l',xlab='t',ylab=expression(x[t]),ylim=range(-0.15,0.15))
points(tv,xf,pch=19,cex=0.5)
legend(0,0.15,c('Final de-trended/de-seasonalized data'),lty=1)

```



```
acf(xf,lag.max=60,main='ACF of residual series')
```

ACF of residual series



We observe more plausibly stationary residual series, although there is some indication of non-constant variance.